Mathematica Bohemica

Milton Braitt; David Hobby; Donald Silberger
Antiassociative groupoids

Mathematica Bohemica, Vol. 142 (2017), No. 1, 27–46

Persistent URL: http://dml.cz/dmlcz/146007

Terms of use:

© Institute of Mathematics AS CR, 2017

Institute of Mathematics of the Czech Academy of Sciences provides access to digitized documents strictly for personal use. Each copy of any part of this document must contain these *Terms of use*.

This document has been digitized, optimized for electronic delivery and stamped with digital signature within the project *DML-CZ: The Czech Digital Mathematics Library* http://dml.cz

ANTIASSOCIATIVE GROUPOIDS

Milton Braitt, Florianópolis,
David Hobby, Donald Silberger, New Paltz

Received February 4, 2015. First published October 31, 2016.
Communicated by Václav Koubek

Abstract. Given a groupoid $\langle G, \star\rangle$, and $k \geqslant 3$, we say that G is antiassociative if an only if for all $x_1, x_2, x_3 \in G$, $(x_1 \star x_2) \star x_3$ and $x_1 \star (x_2 \star x_3)$ are never equal. Generalizing this, $\langle G, \star\rangle$ is k-antiassociative if and only if for all $x_1, x_2, \ldots, x_k \in G$, any two distinct expressions made by putting parentheses in $x_1 \star x_2 \star x_3 \star \ldots \star x_k$ are never equal.

We prove that for every $k \geqslant 3$, there exist finite groupoids that are k-antiassociative. We then generalize this, investigating when other pairs of groupoid terms can be made never equal.

Keywords: groupoid; unification

MSC 2010: 20N02, 08A99, 68Q99, 68T15

1. Introduction

Around fifteen years ago, the second two authors started to investigate finite groupoids which were antiassociative. Instead of obeying the associative law that $(x_1 \star x_2) \star x_3$ and $x_1 \star (x_2 \star x_3)$ are always equal, a groupoid is *antiassociative* if and only if $(x_1 \star x_2) \star x_3$ and $x_1 \star (x_2 \star x_3)$ are never equal. This is a natural change to make to the associative law.

We were aided by a program written by Ming Lei Wu, which went through all the 4^{16} possible 4-element groupoids and returned a list of 421,560 which were antiassociative. About 97 % of these antiassociative groupoids were what we called "deranged", and turned out to be constructible in the following way.

Let G be any set with 2 or more elements. First pick a function $f\colon G \to G$ with the property that $f(x) \neq x$ for all x (the "derangement"). Then define the binary operation on G by $x \star y = f(x)$, or alternatively, by $x \star y = f(y)$. This makes $\langle G, \star\rangle$ a *deranged* groupoid. When $x \star y = f(x)$, we have $(x_1 \star x_2) \star x_3 = f(x_1) \star x_3 =$

DOI: 10.21136/MB.2017.0006-15

$f(f(x_1)) \neq f(x_1) = x_1 \star (x_2 \star x_3)$, showing $\langle G, \star \rangle$ is antiassociative. If $x \star y = f(y)$, the proof is similar. This construction seems to first appear in Example 2.2 of the paper [5] of Drápal and Kepka.

Of the remaining 3 % of the antiassociative groupoids found by the program, almost all had $\star$ tables which were within a few entries of the table of one of the deranged groupoids. But beyond that, we found few patterns in their construction. We conjecture that a similar situation holds for the examples we give in this paper. They probably will not be unique, since it will sometimes be possible to modify them slightly in a haphazard way.

Before moving on to k-antiassociative groupoids, we will invest in some definitions. Using terminology from universal algebra (see [4]), an *algebra* is a set with some number of (finitary) operations on it. A *term* of an algebra is any expression on a finite number of variables that can be made by composing the (basic) operations of the algebra. We will use the same notation both for terms as formal expressions and for the resulting functions on an algebra, since the distinction should be clear from context. This paper will focus on groupoids, which are algebras with a single binary operation. We believe that many of our techniques can be used for algebras with multiple operations of any arity, but will not pursue this avenue here.

An *ordered term* on the variables $x_j, x_{j+1}, x_{j+2}, \ldots, x_{j+k-1}$ is a k-ary term where each variable appears once, in order of their indices. For clarity, we give an inductive definition. Any single variable x_j is a 1-ary ordered term. Now suppose that f is an m-ary basic operation and that $t_1, \ldots, t_m$ are ordered terms on the variables $x_j, x_{j+1}, \ldots, x_{j+n-1}$, respectively. (That is, t_1 is a k_1-ary ordered term on $x_j, x_{j+1}, \ldots, x_{j+k_1-1}$, t_2 is a k_2-ary ordered term on the next k_2 variables, and so on, where $n = k_1 + k_2 + \ldots + k_m$.) Then $f(t_1, t_2, \ldots, t_m)$ is an n-ary ordered term on the variables $x_j, x_{j+1}, \ldots, x_{j+n-1}$. We used ordered terms in groupoids in our earlier papers [3] and [2] and called them *formal products*.

Focusing on groupoids with operation $\star$, we see that there are exactly 5 different ordered terms on the 4 variables x_1, x_2, x_3, x_4. They are: $((x_1 \star x_2) \star x_3) \star x_4$, $(x_1 \star (x_2 \star x_3)) \star x_4$, $(x_1 \star x_2) \star (x_3 \star x_4)$, $x_1 \star ((x_2 \star x_3) \star x_4)$ and $x_1 \star (x_2 \star (x_3 \star x_4))$. As is well known (see [11]), a groupoid has $C(k-1) = (2k-2)!/k!(k-1)!$ many distinct ordered terms on k variables, where $C(m)$ is the m-th Catalan number.

Assume $k \geqslant 3$. Let $s(x_1, \ldots, x_k)$ and $t(x_1, \ldots, x_k)$ be distinct terms of some groupoid $\langle G, \star \rangle$. If $s(x_1, \ldots, x_k) \neq t(x_1, \ldots, x_k)$ for all $x_1, x_2, \ldots, x_k \in G$, then we say that G *separates* s and t. The groupoid $\langle G, \star \rangle$ is *k-antiassociative* if and only if it separates all the distinct pairs of ordered terms on $x_1, x_2, \ldots, x_k$.

Two observations are in order. If G is a groupoid that separates two terms s and t, then every subgroupoid of G also separates s and t. Second, suppose G is a groupoid

that separates s and t, and let H be an arbitrary groupoid (with the same operation symbol). Then the Cartesian product $G \times H$ separates s and t.

There are infinite groupoids that are k-antiassociative for all k. One example is $\langle F^{\sigma}; \odot \rangle$, the set of all formal products under a natural operation which is similar to concatenation. (See [3] for a definition and proof.) The free groupoid (see [4]) on one or more generators is another example, as can be shown by a modification of the proof of Theorem 3.2. (At the end of the proof, where Theorem 3.1 is invoked, one argues directly instead.)

There are no finite groupoids which are k-antiassociative for all k, since the number of k-ary ordered terms increases without bound. Once there are more terms than elements in the groupoid, the Pigeonhole Principle implies that there are terms which will not be separated in the groupoid. This brings us to the following question, which we posed in [2].

Question 1.1. For all $k \geqslant 3$, is there a finite groupoid that is k-antiassociative?

By our observation above, this question may be reduced to the following one.

Question 1.2. For each $k \geqslant 3$ and for all distinct ordered terms s and t on $x_1, x_2, \ldots, x_k$, is there a finite groupoid that separates s and t?

An affirmative answer to the second question gives an affirmative answer to the first. To see this, assume that for all distinct ordered terms s and t on $x_1, x_2, \ldots, x_k$, there is a finite groupoid $G_{s,t}$ that separates s and t. Then the product of these groupoids separates all the k-ary ordered terms, and is k-antiassociative. The other direction is immediate, so the two questions are equivalent.

Note also that whenever $3 \leqslant j < k$, a groupoid $\langle G, \star \rangle$ that is k-antiassociative is also j-antiassociative. For contradiction, suppose $s(x_1, x_2, \ldots, x_j)$ and $t(x_1, x_2, \ldots, x_j)$ are j-ary ordered terms that are not separated in $\langle G, \star \rangle$. We let $r(x_{j+1}, \ldots, x_k)$ be some fixed $(k-j)$-ary ordered term, and form

$$s'(x_1, x_2, \ldots, x_k) = s(x_1, x_2, \ldots, x_j) \star r(x_{j+1}, \ldots, x_k)$$

and

$$t'(x_1, x_2, \ldots, x_k) = t(x_1, x_2, \ldots, x_j) \star r(x_{j+1}, \ldots, x_k).$$

These are two k-ary ordered terms that are not separated in $\langle G, \star \rangle$, a contradiction.

Section 2 will present two preliminary examples. We will answer Question 1.2 in the affirmative in Section 3, and generalize it to arbitrary groupoid terms in Section 4.

2. Preliminary examples

We start with two simple constructions that often yield groupoids separating two distinct k-ary ordered terms. The first is to simply take products of deranged operations. For example, define the operation L_2 on the universe of Z_2 by setting $x\mathrm{L}_2 y = (x{+}1) \bmod 2$. Then we have $(x\mathrm{L}_2 y)\mathrm{L}_2 z = (x{+}2) \bmod 2$, $((x\mathrm{L}_2 y)\mathrm{L}_2 z)\mathrm{L}_2 w = (x{+}3) \bmod 2$, and so on. The value of a term with leftmost variable x is $(x{+}n) \bmod 2$, where n is the depth of x in the term. We also define R_3 on the universe of Z_3 by setting $x\ \mathrm{R}_3\ y = (y+1) \bmod 3$. Similarly, we have that the value of a term with rightmost variable z is $(z+n) \bmod 3$, where n is the depth of z in the term.

We consider the five possible 4-ary ordered terms, which we list as follows:

$$\begin{aligned} t_1 &= ((x_1 \star x_2) \star x_3) \star x_4, \\ t_2 &= (x_1 \star (x_2 \star x_3)) \star x_4, \\ t_3 &= (x_1 \star x_2) \star (x_3 \star x_4), \\ t_4 &= x_1 \star ((x_2 \star x_3) \star x_4), \\ t_5 &= x_1 \star (x_2 \star (x_3 \star x_4)). \end{aligned}$$

In $\langle Z_2, \mathrm{L}_2 \rangle$, we have

$$\begin{aligned} t_1(w,x,y,z) &= (w_1 + 3) \bmod 2, \\ t_2(w,x,y,z) &= (w_1 + 2) \bmod 2, \\ t_3(w,x,y,z) &= (w_1 + 2) \bmod 2, \\ t_4(w,x,y,z) &= (w_1 + 1) \bmod 2, \\ t_5(w,x,y,z) &= (w_1 + 1) \bmod 2, \end{aligned}$$

so all the terms in $\{t_1, t_4, t_5\}$ are separated from those in $\{t_2, t_3\}$ in this groupoid. Similarly, the terms in the sets $\{t_1, t_2\}$, $\{t_3, t_4\}$ and $\{t_5\}$ are all separated from those in the other sets in the groupoid $\langle Z_3, \mathrm{R}_3 \rangle$. Continuing, all five terms are separated from each other in the product of the two groupoids.

The problem with this approach is that the value of a term only depends on the depths of its leftmost and rightmost variables, so terms that have those two variables at the same depth cannot be separated this way.

The next construction partially avoids this problem. Suppose that $A = \langle A, + \rangle$ is an abelian group, that α and β are endomorphisms of $\langle A, + \rangle$, and that c is a fixed element of A. We define an operation $\star$ on A by setting $x \star y = \alpha(x) + \beta(y) + c$, and call the groupoid $\langle A, \star \rangle$ the *affine endomorphism groupoid* for A, α, β and c. We denote this groupoid by $E(A, \alpha, \beta, c)$.

As an example, suppose we want an affine endomorphism groupoid that separates the terms $s(v,w,x,y,z) = ((v \star w) \star (x \star y)) \star z$ and $t(v,w,x,y,z) = ((v \star (w \star x)) \star y) \star z$. In both terms, v has depth 3 and z has depth 1, so the previous approach cannot succeed.

In $E(A, \alpha, \beta, c)$, we get

$$\begin{aligned} s(v,w,x,y,z) &= ((\alpha(v) + \beta(w) + c) \star (\alpha(x) + \beta(y) + c)) \star z \\ &= (\alpha^2(v) + \alpha\beta(w) + \alpha(c) + \beta\alpha(x) + \beta^2(y) + \beta(c) + c) \star z \\ &= \alpha^3(v) + \alpha^2\beta(w) + \alpha^2(c) + \alpha\beta\alpha(x) + \alpha\beta^2(y) + \alpha\beta(c) \\ &\quad + \alpha(c) + \beta(z) + c. \end{aligned}$$

This is quite messy, so we make the simplifying assumptions that $\alpha^3 = \alpha^2$, that $\beta^2 = \beta$, and that α and β commute. This gives us

$$\begin{aligned} s(v,w,x,y,z) = \alpha^2(v) + \alpha^2\beta(w) + \alpha^2\beta(x) + \alpha\beta(y) + \beta(z) \\ + \alpha^2(c) + \alpha\beta(c) + \alpha(c) + c. \end{aligned}$$

And a similar calculation gives

$$\begin{aligned} t(v,w,x,y,z) = \alpha^2(v) + \alpha^2\beta(w) + \alpha^2\beta(x) + \alpha\beta(y) + \beta(z) \\ + \alpha^2\beta(c) + \alpha^2(c) + \alpha(c) + c. \end{aligned}$$

Observe that both terms have the identical portion $\alpha^2(v) + \alpha^2\beta(w) + \alpha^2\beta(x) + \alpha\beta(y) + \beta(z)$, and only differ in their constants. (Our choice of simplifying assumptions was designed to do this.) So we can separate the terms by ensuring that $\alpha^2(c) + \alpha\beta(c) + \alpha(c) + c$ and $\alpha^2\beta(c) + \alpha^2(c) + \alpha(c) + c$ have different values.

Fortunately, there are A, α, β and c that satisfy these conditions. We may work over $\mathbb{Z}_2$, and consider 2×3 matrices with elements in $\mathbb{Z}_2$. This gives us that the group A is isomorphic to $\mathbb{Z}_2^6$, a 64-element group. The desired actions of α and β on A can be realized by letting β copy the top row of A onto the bottom row, and by letting α copy the left column of A onto the middle column and the middle column onto the right column. That is,

$$\alpha \begin{bmatrix} d & e & f \\ g & h & i \end{bmatrix} = \begin{bmatrix} d & d & e \\ g & g & h \end{bmatrix} \quad \text{and} \quad \beta \begin{bmatrix} d & e & f \\ g & h & i \end{bmatrix} = \begin{bmatrix} d & e & f \\ d & e & f \end{bmatrix}.$$

Finally, we take

$$c = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

This gives

$$\alpha\beta(c) = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix} \quad \text{and} \quad \alpha^2\beta(c) = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix},$$

so

$$s(\vec{0}) = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix} \quad \text{and} \quad t(\vec{0}) = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix}.$$

The above technique requires making assumptions about α and β in order to simplify the expressions for the terms. One has some latitude with the assumptions. For example, one may take $\alpha^{k+1} = \alpha^k$, or $\beta^{k+1} = \beta^k$ for any value of k, and no longer require that α and β commute. But a point is reached where that no longer helps. We were unable to use the above method to produce a groupoid that separated the two 5-ary terms $s = (x_1 \star (x_2 \star x_3)) \star (x_4 \star x_5)$ and $t = (x_1 \star x_2) \star ((x_3 \star x_4) \star x_5)$. (These terms are represented by trees in Figure 1.)

We note in passing that if $\alpha\beta = \beta\alpha$, then

$$\begin{aligned}(w \star x) \star (y \star z) &= \alpha^2(w) + \alpha(c) + \alpha\beta(x) + \beta\alpha(y) + \beta(c) + \beta^2(z) \\ &= \alpha^2(w) + \alpha(c) + \alpha\beta(y) + \beta\alpha(x) + \beta(c) + \beta^2(z) = (w \star y) \star (x \star z).\end{aligned}$$

So if α and β commute, the groupoid satisfies the law $(w\star x)\star(y\star z) \approx (w\star y)\star(x\star z)$. While this is not a generalized associative law, it is known as the *medial* or *entropic* law, and has been extensively studied. Although there are now many articles on specializations of this law, the best general survey seems to be the 1983 paper [8] by Ježek and Kepka.

So we turn to another method, which we will present in the next section.

3. Finite k-antiassociative groupoids

We will use a somewhat involved construction, and will require some preliminary definitions. Recall that a *full binary tree* is a rooted tree where every internal node has exactly two children. (For further definitions and theorems, see [9] or a recent text in discrete mathematics or data structures.)

When full binary trees are used as data structures, the two nodes directly below each internal node are called its *left* and *right* children, and the subtrees with these children as roots are the *left* and *right subtrees* of that node. As is well known, groupoid terms correspond to full binary trees with leaves labeled by variables. If s is a groupoid term, we will denote the corresponding tree by $T(s)$. This correspondence may be defined recursively as follows. If s is a single variable x_i, then $T(s)$ is a tree

with one node, labeled x_i. If s and t are groupoid terms, then $T(s \star t)$ is the tree with a root that has $T(s)$ as its left subtree and $T(t)$ as its right subtree.

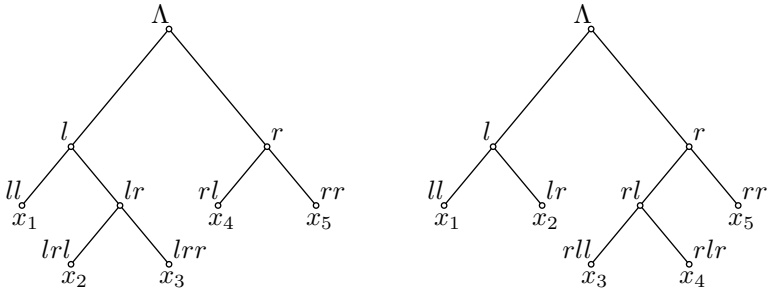
We will also label the nodes of binary trees with strings made from the characters 'l' and 'r'. As is usual, we will write the set of all such strings as $\{l, r\}^*$. In dealing with strings, we will show concatenation by simply writing the two strings next to each other. We use Λ to denote the empty string, which is the identity for concatenation. Our labeling may be defined recursively as follows.

The root is labeled Λ. If a node is labeled a, then its left and right children are labeled al and ar, respectively. These labels may be thought of as directions for how to get to a node by starting at the root and turning the correct way at each branching.

Given a string p, an *initial substring* of p is a string q such that $p = qu$ for some string u. (This is sometimes called a *prefix* in the literature. Note that the empty string Λ is an initial substring of every string.) A substring is *proper* if it is not equal to the entire original string, and *nontrivial* if it is not equal to Λ.

Putting these two ideas together, occurrences of variables in a groupoid term s correspond to leaves of $T(s)$. The string that is the label of the leaf corresponding to an occurrence of the variable x_i will be called the *path* of that occurrence. If x_i only occurs once, we may also call this the path of x_i. Generalizing this, for any subterm b of s, we have that the *path* of b is also the label of the interior node of $T(s)$ corresponding to the root of subtree $T(b)$.

For example, consider $s = (x_1 \star (x_2 \star x_3)) \star (x_4 \star x_5)$. We have $\text{path}(x_1) = ll$, $\text{path}(x_2) = lrl$, $\text{path}(x_3) = lrr$, $\text{path}(x_4) = rl$, $\text{path}(x_5) = rr$ and $\text{path}(x_2 \star x_3) = lr$. (When there is danger of confusion, we will write $\text{path}_s(x_i)$ to show we mean the path in the term s.) The tree for this term is on the left side of Figure 1.



$(x_1 \star (x_2 \star x_3)) \star (x_4 \star x_5)$ $\qquad$ $(x_1 \star x_2) \star ((x_3 \star x_4) \star x_5)$

Figure 1. Trees for two terms.

If s is a groupoid term, we use $\text{Paths}(s)$ for the set of all paths to variables in s. Similarly, we have paths to the internal nodes of the tree $T(s)$; these correspond to

proper initial substrings of paths to the leaves of $T(s)$. Given the term s with q the path to a node of $T(s)$, we let s_q denote the subterm of s with $T(s_q)$ rooted at the node q of $T(s)$. Then for any subterm b of s, if we let q be the path of b in s, we have $b = s_q$.

Our long-term goal is to form a groupoid that separates any two distinct k-ary ordered terms s and t. We will need some preliminary ideas in order to do this. Our groupoids will be finite homomorphic images of an infinite groupoid with universe $\mathbb{Z}_2^{\mathbb{N}}$, where $\mathbb{Z}_2$ is the 2-element field. So the elements of the infinite groupoid will be vectors with index set $\mathbb{N} = \{0, 1, 2, \ldots\}$. The groupoid operation $\star$ will be constructed below in such a way that there exists an integer M such that for all $u, v \in \mathbb{Z}_2^{\mathbb{N}}$, the j-th components of $u \star v$ will be zero for all $j > M$. Any two terms that are separated by the infinite groupoid on $\mathbb{Z}_2^{\mathbb{N}}$ will then be separated by its finite projection onto the first M coordinates. We will usually leave this final reduction to a finite groupoid to the reader.

We will actually be using only the additive structure of the field $\mathbb{Z}_2$, and viewing it as an abelian group. Our groupoids will all be affine endomorphism groupoids, although the endomorphisms will be built up from their actions on the components of vectors. One nice consequence of this is that we will be able to add groupoid operations pointwise. If $\star_1$ and $\star_2$ are two groupoid operations on vectors in $\mathbb{Z}_2^{\mathbb{N}}$, their sum $\star_1 + \star_2$ will be defined by $\vec{x}\,(\star_1 + \star_2)\,\vec{y} = (\vec{x}\star_1\vec{y}) + (\vec{x}\star_2\vec{y})$. Since we are working over $\mathbb{Z}_2$, all additions of values such as the above are done modulo 2. We will periodically note this fact, but not always.

We will define groupoid operations by their actions on components. In this section we will use the convention that the vectors x, y and z are such that $z = x \star y$ for our groupoid operation $\star$. We will also simply write x instead of $\vec{x}$, and write $x[a]$ for the a-th component of the vector x. (For clarity, we will always use square brackets for this.) To specify a groupoid operation, it then suffices to say what $z[i]$ is for all i. We will do this by giving a sequence of equations for the $z[i]$. To emphasize that values are being assigned to the $z[i]$, we will use $:=$ instead of the normal equality symbol. One further convention is that each $z[i]$ will be zero, unless that $z[i]$ is explicitly assigned a value.

For example, consider the groupoid operation which we will later call $\|2, lr, 0\|$. We define it by the two equations $z[0] := x[a]$ and $z[a] := y[2]$. Here, a is some index disjoint from $\{0, 2\}$, we take $a = 1$. Writing our operation as $\star$, we have $\langle x[0], x[a], x[2], \ldots\rangle \star \langle y[0], y[a], y[2], \ldots\rangle = \langle x[a], y[2], 0, 0, 0, \ldots, \rangle$. Continuing to use $\star$ for this operation, consider the term $s = (u \star v) \star w$. We have $u \star v = \langle u[a], v[2], 0, 0, 0, \ldots\rangle$, and $(u \star v) \star w = \langle u[a], v[2], 0, 0, 0, \ldots\rangle \star \langle w[0], w[a], w[2], \ldots\rangle = \langle v[2], w[2], 0, 0, 0, \ldots\rangle$. The 0-th component of s is the 2nd component of v, where

$\text{path}_s(v) = lr$. This motivates calling the operation $\|2, lr, 0\|$, since it takes the 2nd component in the subterm s_{lr} to the 0-th component of s.

When using the operation $\|2, lr, 0\|$, we will be looking only at the 0-th component of the output, and ignoring the a-th component. With this understanding, it makes little difference what the index a is. So we will assume that indices such as a, b and so on are always chosen to minimize *collisions*. This means that no indices will be equal unless they are explicitly represented with equivalent expressions. This can be easily achieved by appropriate choices of values for a, b and so on, and will not jeopardize the finiteness of any groupoids we produce. As long as there are no collisions, the finite groupoids obtained via the projection homomorphism for different values of a will be isomorphic. Accordingly, we will speak of *the* groupoid operation $\|2, lr, 0\|$, and so on.

Definition 3.1. Let $p = p_0p_1p_2 \dots p_j$ be a nonempty string in $\{l, r\}^*$, and let m and n be natural numbers. Then the operation $\|m, p, n\|$ is defined via the following equations, where we assume that $a, a+1, \dots, a+j-1$ are distinct from m and n. If $j = 0$, we set $z[n] := x[m]$ if $p_0 = l$, and $z[n] := y[m]$ if $p_0 := r$. If $j > 0$, we proceed as follows.

If p_0 is l, the first equation is $z[n] := x[a]$, and if p_0 is r, it is $z[n] := y[a]$. If $p_1 = l$, the next equation is $z[a] := x[a+1]$, and if $p_1 = r$, it is $z[a] := y[a+1]$. This pattern continues, with $z[a+i] := x[a+i+1]$ if $p_{i+1} = l$, or $z[a+i] := y[a+i+1]$ if $p_{i+1} = r$, for all $i \leqslant j-2$. The last equation is $z[a+j-1] := x[m]$ if $p_j = l$, and it is $z[a+j-1] := y[m]$ if $p_j = r$.

The idea is that $\|m, p, n\|$ transfers the value of the m-th component of the vector with path p in the term s to the n-th component of the result of s. Here is a more detailed example to illustrate this definition.

Let s be $(((v_1 \star v_2) \star v_3) \star v_4) \star (v_5 \star v_6)$, so the path of v_2 is $lllr$. We let $n = 0$ and $m = 4$. The operation $\|4, lllr, 0\|$ will take the value of $v_2[4]$ and assign it to $s[0]$. In the definition, we have $p = lllr = p_0p_1p_2p_3$, and $j = 3$. We let $a = 1$, so $a + 1 = 2$, and $a + j - 1 = 1 + 3 - 1 = 3$.

Then $p_0 = l$, giving $z[0] := x[1]$. Next, $p_1 = l$, giving $z[1] := x[2]$. Continuing, $p_2 = l$ and $z[2] := x[3]$. Finally, $z[3] := y[4]$ since p_3 is r.

Thus we have $z = \langle z[0], z[1], z[2], z[3], z[4], \dots \rangle = \langle x[1], x[2], x[3], y[4], 0, 0, \dots \rangle$, when $z = x \star y$. Letting the operation $\star$ be $\|4, p, 0\|$ does make $s[0] = v_2[4]$. When $s = (((v_1 \star v_2) \star v_3) \star v_4) \star (v_5 \star v_6)$, we have that the output z is s and the left input x is $((v_1 \star v_2) \star v_3) \star v_4$. Since $z[0] := x[1]$, we have $s[0] = ((v_1 \star v_2) \star v_3) \star v_4[1]$. Continuing in this fashion, we get $s[0] = ((v_1 \star v_2) \star v_3) \star v_4[1] = (v_1 \star v_2) \star v_3[2] = v_1 \star v_2[3] = v_2[4]$.

We want our functions $\|m, p, n\|$ to have as few side effects as possible. To accomplish this, we are assuming throughout that none of the indices used to define

$\|m,p,n\|$ is equal to any of the others, except that possibly $m = n$. In other words, the operation $\|m,p,n\|$ is *duplicate-free*. If m_1 is distinct from both m_2 and m_0, and p and q are strings in $\{l,r\}^*$, then the operation $\|m_2,q,m_1\| + \|m_1,p,m_0\|$ is duplicate-free by our convention that indices are chosen to minimize collisions. In isolation, the sum $\|m_2,q,m_1\| + \|m_1,p,m_0\|$ is equivalent to $\|m_2,pq,m_0\|$. The one difference is that the former explicitly mentions the index m_1. We will henceforth assume that all our groupoid operations are duplicate-free.

Lemma 3.1. *Let $\star$ be a duplicate- and collision-free groupoid operation that contains $\|m,p,n\|$ as a summand, and let s be a groupoid term where p is the path to a node of $T(s)$. Letting s_p be the subterm of s at that node, $s[n] = s_p[m]$ for all values of the variables of s.*

Proof. Since $\star$ is duplicate- and collision-free, the only summand of $\star$ that affects the value of $s[n]$ is $\|m,p,n\|$. So we may ignore the rest of $\star$, and assume $\star$ is $\|m,p,n\|$. Writing $p = p_0p_1p_2\ldots p_j$ where the p_i are r or l, we will prove the lemma by induction on j. Our basis is when $j = 0$, making the operation $\|m,p_0,n\|$. We will show the case where $p_0 = l$, the one for $p_0 = r$ is similar. Now $s = s_l \star s_r$, where $\star$ is $\|m,l,n\|$. The one relevant assignment is $z[n] := x[m]$, giving $s[n] = z[n] = x[m] = s_l[m]$, as desired.

For the induction step, assume the statement is true for $j-1$, and that we want to show it for the path $p = p_0p_1p_2\ldots p_j$. We write $\star = \|m,p,n\|$ as $\|m,p_j,b\| + \|b,p_0p_1p_2\ldots p_{j-1},n\|$ for some new index b, and let q be $p_0p_1\ldots p_{j-1}$, so $p = qp_j$. By the statement for $j-1$, $s[n] = s_q[b]$. We have $s_q[b] = (s_{ql} \star s_{qr})[b] = (s_{ql}\,\|m,p_j,b\|\,s_{qr})[b]$, where the last step follows because indices are chosen to minimize collisions. There are now two cases. We will show the one for $p_j = r$; the case for $p_j = l$ is similar. Since $p_j = r$, we have $z[b] := y[m]$ in $\|m,p_j,b\|$. So $s_q[b] = s_{qr}[m] = s_p[m]$, since $qr = qp_j = p$. Thus $s[n] = s_q[b] = s_p[m]$, as desired. □

Given the groupoid operation $\|m,p,n\|$, we define the *tweaked* operation $\|m,p,n\|'$ to be identical to $\|m,p,n\|$ except for one assignment. Writing p as p_0q, $\|m,p,n\|$ has an assignment of the form $z[n] := x[k]$ if $p_0 = l$ and one of the form $z[n] := y[k]$ if $p_0 = r$. Whichever one occurs, we modify it by adding 1, giving $z[n] := (x[k]+1) \bmod 2$ if $p_0 = l$ or giving $z[n] := (y[k]+1) \bmod 2$ if $p_0 = r$.

A slight modification of the proof of the previous lemma then establishes the following.

Lemma 3.2. *Let $\star$ be a duplicate- and collision-free groupoid operation that contains $\|m,p,n\|'$ as a summand, and let s be a groupoid term where p is the*

path to a node of $T(s)$. *Letting* $t = s_p$ *be the subterm of* s *at that node,* $s[n] = (t[m] + 1) \bmod 2$.

We are now ready to establish a powerful theorem, which holds for all groupoid terms regardless of any conditions on the order or number of appearances of variables.

Theorem 3.1. *Let* s *and* t *be any groupoid terms. Suppose that the variable* x *has an occurrence in* s *where the path to that occurrence is* p, *and that* x *has an occurrence in* t *where the path to that occurrence is* q. *Then if* q *is a proper initial substring of* p, *the terms* s *and* t *can be separated.*

P r o o f. Let s, t, x, p and q be as above. By hypothesis, $p = qw$ for a nonempty string w. We let $\star$ be $\|1, q, 0\| + \|1, w, 1\|'$.

First consider the value of $t[0]$ for this $\star$. Since $\|1, w, 1\|'$ does not have an assignment to $z[0]$, $\|1, w, 1\|'$ makes $t[0] = 0$, and we can ignore it. As for $\|1, q, 0\|$, Lemma 3.1 gives $t[0] = t_q[1] = x[1]$. This implies that $\star$ sets $t[0] = x[1]$.

Now consider the value of $s[0]$ for the above $\star$. As in our calculation for $t[0]$, we have $s[0] = s_q[1]$. But now s_q is a nontrivial subterm of s, so we compute $s_q[1]$. The operation $\|1, q, 0\|$ has no effect on $s_q[1]$, so we ignore it and just consider the effect of $\|1, w, 1\|'$. It gives $s_q[1] = s_{qw}[1] + 1$, by Lemma 3.2. Putting these together, we have $s[0] = s_q[1] = s_{qw}[1] + 1 = s_p[1] + 1 = x[1] + 1$. This shows that s and t always have different values in a finite groupoid, since it is always true that $s[0] \neq t[0]$. □

Theorem 3.2. *For all* $k \geqslant 3$ *there is a* k*-antiassociative finite groupoid.*

P r o o f. It is enough to produce a finite groupoid that separates any two distinct k-ary ordered terms s and t. Given any two distinct terms s and t with $k \geqslant 3$, we let x_m be the leftmost variable on which s and t do not agree, in the sense that $\mathrm{path}_s(x_i) = \mathrm{path}_t(x_i)$ for all $i < m$, and $\mathrm{path}_s(x_m) \neq \mathrm{path}_t(x_m)$.

We claim that for any two such distinct k-ary terms s and t, one of $\mathrm{path}_s(x_m)$ or $\mathrm{path}_t(x_m)$ is a proper initial substring of the other. To see this, suppose we have a counterexample consisting of terms s and t where x_m is the leftmost variable on which they disagree but neither $\mathrm{path}_s(x_m)$ nor $\mathrm{path}_t(x_m)$ is a proper initial substring of the other. Letting j be the minimum of the lengths of $\mathrm{path}_s(x_m)$ and $\mathrm{path}_t(x_m)$, we assume that our counterexample has the least possible value of j.

If our counterexample has $j = 0$, then either $s = x_m$ or $t = x_m$. Without loss of generality, assume $s = x_m$. Then $\mathrm{path}_s(x_m) = \Lambda$. If $\mathrm{path}_t(x_m)$ is also Λ, we have $s = x_m = t$, a contradiction. So $\mathrm{path}_t(x_m) \neq \Lambda$, and $\mathrm{path}_t(x_m)$ has $\mathrm{path}_s(x_m)$ as a proper initial substring. This shows our minimal counterexample must have $j > 0$.

Then $s = s_l \star s_r$ and $t = t_l \star t_r$ for some terms s_l, s_r, t_l and t_r. We have two cases, depending on where x_m occurs.

If x_m occurs in s_r, then x_m also occurs in t_r since $s_l = t_l$ because s and t agree for all $i < m$. But then x_m is the leftmost variable on which s_r and t_r disagree. Since s and t form a minimal counterexample, s_r and t_r are not a counterexample and so one of $\mathrm{path}_{s_r}(x_m)$ and $\mathrm{path}_{t_r}(x_m)$ is a proper initial substring of the other. Since $\mathrm{path}_s(x_m)$ is r followed by $\mathrm{path}_{s_r}(x_m)$ and $\mathrm{path}_t(x_m)$ is r followed by $\mathrm{path}_{s_r}(x_m)$, one of them is also a proper initial substring of the other, a contradiction.

So x_m occurs in s_l. As in the previous paragraph, if x_m occurred in t_r, we would get that x_m occurred in s_r. Thus x_m occurs in t_l. Then a similar argument also yields a contradiction. This proves the claim.

Now let distinct k-ary s and t with $k \geqslant 3$ be given. The claim gives us that one of $\mathrm{path}_t(x_m)$ and $\mathrm{path}_s(x_m)$ is a proper initial substring of the other. We apply Theorem 3.1, and obtain a finite groupoid that separates s and t. □

4. Separating arbitrary groupoid terms

We can generalize the questions of the previous section, by relaxing the condition that each variable appears once in every term in order of their indices.

As before, we can reduce everything to the problem of finding finite algebras that separate pairs of terms. We would like to have a nice characterization of which pairs of groupoid terms can be separated in a finite groupoid. So we will also investigate when it is impossible to separate a pair of terms in any groupoid.

We need a bit of preliminary material on free algebras. A more detailed exposition may be found in [4]. We use $\mathbf{G}$ for the class of all groupoids, and let $F_{\mathbf{G}}(y_0, y_1, \ldots, y_{n-1})$ denote the free groupoid with generators $y_0, y_1, \ldots, y_{n-1}$. The key feature of $F_{\mathbf{G}}(y_0, y_1, \ldots, y_{n-1})$ is that it has the Universal Mapping Property for the class of groupoids. That is, if G is any groupoid with elements $g_0, g_1, \ldots, g_{n-1}$, then there is a unique homomorphism φ from $F_{\mathbf{G}}(y_0, y_1, \ldots, y_{n-1})$ into G where $\varphi(y_i) = g_i$ for all i.

Theorem 4.1. *Let s and t be groupoid terms, each on a set of variables that is a subset of $\{y_0, y_1, \ldots, y_{n-1}\}$. Then the following are equivalent:*
(1) *s and t are separated in some groupoid,*
(2) *s and t are separated in $F_{\mathbf{G}}(y_0, y_1, \ldots, y_{n-1})$,*
(3) *s and t are separated in $F_{\mathbf{G}}(x)$, the free groupoid on one variable.*

Proof. Let s and t be groupoid terms with all their variables in $\{y_0, y_1, \ldots, y_{n-1}\}$. It is clear that (2) implies (1). To see that (3) implies (2), suppose that (2) fails. Then there are terms $h_0, h_1, \ldots, h_{n-1}$ in $F_{\mathbf{G}}(y_0, y_1, \ldots, y_{n-1})$ with $s(h_0, h_1, \ldots, h_{n-1}) = t(h_0, h_1, \ldots, h_{n-1})$. The h_i are all generated from $\{y_0, \ldots,$

$y_{n-1}\}$ by repeatedly using the groupoid operation. Now consider the homomorphism φ from $F_{\mathbf{G}}(y_0, y_1, \ldots, y_{n-1})$ into $F_{\mathbf{G}}(x)$ that takes all of the y_i to x. Denoting the image of each h_i by h'_i, we have that $s(h'_0, h'_1, \ldots, h'_{n-1}) = t(h'_0, h'_1, \ldots, h'_{n-1})$ in $F_{\mathbf{G}}(x)$, so (3) fails.

To see that (1) implies (3), assume that (3) fails. So we have $f_0, f_1, \ldots, f_{n-1} \in F_{\mathbf{G}}(x)$ with $s(f_0, f_1, \ldots, f_{n-1}) = t(f_0, f_1, \ldots, f_{n-1})$. Letting G be any groupoid, we pick any $c \in G$, and consider the homomorphism φ from $F_{\mathbf{G}}(x)$ to G that takes x to c. Letting the image of each f_i be f'_i, we have that $s(f'_0, f'_1, \ldots, f'_{n-1}) = t(f'_0, f'_1, \ldots, f'_{n-1})$ in G, so (1) fails. □

The free groupoid $F_{\mathbf{G}}(x)$ is easy to work with, since all of its elements may be viewed as groupoid terms in the single variable x. Terms s and t are separated in $F_{\mathbf{G}}(x)$ if and only if there are no terms $f_0(x), f_1(x), \ldots, f_{n-1}(x) \in F_{\mathbf{G}}(x)$ that can be substituted for the variables of s and t to yield $s(f_0(x), f_1(x), \ldots, f_{n-1}(x)) = t(f_0(x), f_1(x), \ldots, f_{n-1}(x))$.

This relates to the notion of *unification* of terms, which has been extensively studied in computer science. The introduction of the topic was by Herbrand, in [6]. Modern work was pioneered by Robinson, in [10]. A good survey article is by Baader and Snyder (in [1]). Consider two terms $s(x_0, \ldots x_{m-1})$ and $t(y_0, \ldots y_{n-1})$. The terms are *unifiable* if there are terms $r_0, \ldots, r_{m-1}$ and $u_0, \ldots u_{n-1}$ such that substituting the r_i for the x_i in s and the u_j for the y_j in t makes the two resulting terms identical. This identical term is a *unifier* and the corresponding substitution is a *unification*. In other words, the terms s and t can be unified if and only if they cannot be separated in a free algebra. In view of the previous theorem, two terms cannot be unified if and only if there is a groupoid where they are separated.

While much work on unification is concerned with algorithms, we will follow the more abstract approach in [1], which was first presented by Huet in [7]. As in his Definition 2.11, we consider equivalence relations on groupoid terms, which we call *term relations*. (A term relation is said to be *homogeneous* if terms $f(\ldots)$ and $g(\ldots)$ are never equivalent for distinct operation symbols f and g. All term relations on groupoid terms are of course homogeneous, so we modify the standard definition to omit this condition.) A term relation is *acyclic* if no term is equivalent to one of its proper subterms. This leads to the following definition.

Definition 4.1. A groupoid term relation $\equiv$ is a *unification relation* if and only if it is acyclic and satisfies the following *unification axiom*: For all terms s, t, u and v, $s \star t \equiv u \star v$ implies $s \equiv u$ and $t \equiv v$.

Referring again to [1] for the details, we have that terms s and t can be unified if and only if there is a unification relation $\equiv$ with $s \equiv t$. If there is such a unification relation, then there is a unique minimal one, the *unification closure* of s and t. If s

and t can be unified, they also have a *most general unifier* or mgu, where any unifier of s and t can be obtained from their mgu by uniformly substituting terms for its variables. This most general unifier is unique up to renaming its variables, and can be easily constructed from the unification closure of s and t.

For example, consider $s = (x \star y) \star (z \star y)$ and $t = z \star ((x \star y) \star (x \star x))$. We will attempt to construct their unification closure $\equiv$ and the corresponding mgu. We must have $s \equiv t$, and start with this. Using the unification axiom, we obtain $x \star y \equiv z$ and $z \star y \equiv (x \star y) \star (x \star x)$. Applying the unification axiom again to the last equivalence, we get $z \equiv x \star y$ (a duplicate) and $y \equiv x \star x$. The non-singleton classes of $\equiv$ that contain variables are now $\{y, x \star x\}$ and $\{z, x \star y\}$, where x is in a class by itself.

To construct the mgu, we pick a representative of each class, where we must pick a nonvariable term if there is one in the class. Letting u, v and w be arbitrary terms, we let $\varsigma(w)$ be the representative of the class of w. In our example, this gives $\varsigma(y) = x \star x$, $\varsigma(z) = x \star y$, and $\varsigma(x) = x$. Now we recursively define the function σ from terms to terms by letting $\sigma(w)$ be $\varsigma(w)$ if $\varsigma(w)$ is a variable, and letting $\sigma(w)$ be $\sigma(u) \star \sigma(v)$ if $\varsigma(w)$ is $u \star v$. In our example, this gives

$$\begin{aligned}\sigma(s) &= \sigma(x \star y) \star \sigma(z \star y) = (\sigma(x) \star \sigma(y)) \star (\sigma(z) \star \sigma(y)) \\ &= (x \star (\sigma(x) \star \sigma(x))) \star ((\sigma(x) \star \sigma(y)) \star (\sigma(x) \star \sigma(x))) \\ &= (x \star (x \star x)) \star ((x \star (x \star x)) \star (x \star x)),\end{aligned}$$

where the last term is the mgu of s and t.

If we try to separate the two groupoid terms $s(x, y) = x \star y$ and $t(x, y) = y \star x$, we rapidly run into trouble. When $x = y$, both terms reduce to their mgu, $x \star x$, so it is impossible to separate them in any groupoid. This trick of identifying variables can be applied whenever s and t have the same *shape*, which we can define rigorously as follows. Let χ be a distinguished variable symbol, which we agree to use nowhere else. Then we simply define the *shape* of a term $s(x_1, x_2, \ldots, x_k)$ to be the term $s(\chi, \chi, \ldots, \chi)$.

As an aside, note that we can easily make the term functions $x \star y$ and $y \star x$ not equal whenever $x \neq y$, for instance by letting $\star$ be $-$ over Z_3. This prompts the following question, which we will not deal with further in this paper.

Question 4.1. Suppose that s and t are two terms of the same shape, and let $x_1, x_2, \ldots, x_k$ be all the variables appearing in either of them. View s and t as operations on all these variables, whether or not they actually appear. Let $\equiv$ be the unification closure of s and t, and for any finite set A let R_A be the k-ary relation on A defined by $\langle y_1, y_2, \ldots, y_k \rangle \in R_A$ if and only if $y_i = y_j$ for all $i, j \leqslant k$ with

$x_i \equiv x_j$. Thus $\langle y_1, y_2, \ldots, y_k \rangle \in R_A$ forces $s(y_1, y_2, \ldots, y_k) = t(y_1, y_2, \ldots, y_k)$ on A, regardless of how the groupoid operation is defined on A.

For which s and t is there a groupoid with finite universe A, where

$$s(y_1, y_2, \ldots, y_k) \neq t(y_1, y_2, \ldots, y_k) \quad \text{whenever } \langle y_1, y_2, \ldots, y_k \rangle \notin R_A?$$

From now on, we will focus on separating two groupoid terms of different shapes. Since we are now dealing with arbitrary terms, variables may occur more than once in a given term. For clarity, we will usually use primes to distinguish occurrences of a variable from the variable itself, so that x' might denote some particular occurrence of x. We will say that terms s and t are *finitely separated* whenever they are separated in some finite groupoid.

Observe that any groupoid term s has a *natural order* to the occurrences of its variables, the order produced by an inorder transversal of the leaves of its full binary tree $T(s)$. We will always write terms by listing occurrences of variables in this natural order. In this case, we call x'_1 the *leftmost* variable occurrence in $s(x_1, \ldots)$. Each variable occurrence in s corresponds to a leaf in $T(s)$, so occurrences of a given variable may be distinguished by their paths in $T(s)$. The leftmost variable occurrence in s is then the only one with a path in $\{l\}^*$.

By the *depth* of an occurrence of a variable in the term s, we mean its height in $T(s)$. We will denote the depth in s of the variable occurrence x' by $d_s(x')$. Note that this is the same as the length of the string $\mathrm{path}_s(x')$.

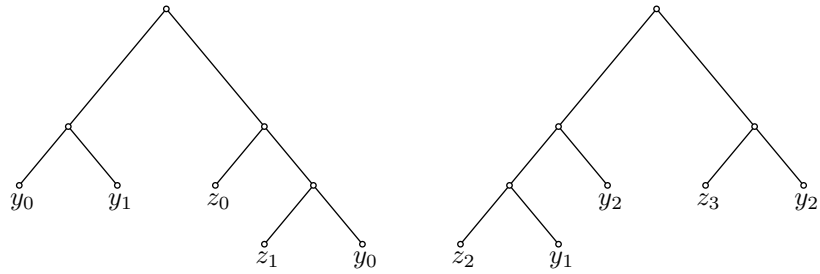
A naive intuition would be that terms s and t could not be separated when there were a number of variables occurring in one term and not the other. It is certainly true that having more variables of this sort gives more possibilities to assign values to them that would unify s and t. For example, let s be $(x \star y) \star z$, and let t be $(x \star x) \star (x \star x)$. Then substituting x for y and $x \star x$ for z in s, it becomes $(x \star x) \star (x \star x)$, which is t. So s and t cannot be separated in any groupoid.

However, there are terms with only a single variable in common that can still be separated in a finite groupoid. For example, let s be $x \star p$ and let t be $(x \star y) \star q$, where p and q can be arbitrary terms on any variables. For the leftmost occurrences of x, we have $\mathrm{path}_s(x) = l$ and $\mathrm{path}_t(x) = ll$. So Theorem 3.1 gives a finite groupoid that separates s and t. Looking at this in terms of constructing the unification closure $\equiv$ of s and t, we have $x \star y \equiv x$ by the unification axiom, so $\equiv$ is not acyclic, showing that s and t cannot be unified.

To continue our investigation, we need the following extension of Theorem 3.1, which requires further definitions to state. If s and t are groupoid terms and y and z are variables, we say that y *occurs above* z if there are occurrences y' of y and z' of z such that either $\mathrm{path}_s(y')$ is an initial substring of $\mathrm{path}_t(z')$ or $\mathrm{path}_t(y')$ is an initial

substring of $\mathrm{path}_s(z')$. In this situation, we also say that the occurrence y' is *above* the occurrence z'. Similarly, y *occurs strictly above* z if there are occurrences y' of y and z' of z such that either $\mathrm{path}_s(y')$ is a proper initial substring of $\mathrm{path}_t(z')$ or $\mathrm{path}_t(y')$ is a proper initial substring of $\mathrm{path}_s(z')$.

We say that terms s and t *have a cycle* if there is a sequence of variables $y_0, y_1, \ldots, y_{m-1}$ where y_0 occurs above y_1, y_1 occurs above y_2, and so on, ending with y_{m-1} occurring above y_0, where at least one of these occurrences is strictly above the other. The hypothesis of Theorem 3.1 is that a single variable x occurs above itself, so that s and t have a cycle of length 1, where the sequence $y_0, y_1, \ldots, y_{m-1}$ is just x. Our next theorem extends this result to cycles of arbitrary length.



$s = (y_0 \star y_1) \star (z_0 \star (z_1 \star y_0))$ $\qquad$ $t = ((z_2 \star y_1) \star y_2) \star (z_3 \star y_2)$

Figure 2. Two terms with a cycle.

The proof of the next theorem will be easier to follow if we have an example for reference. It may be useful to refer back to this example while reading the proof, as some of the notation it uses is defined in the proof. Figure 2 shows a cycle $y_0 y_1 y_2$ of length 3, where $s = (y_0 \star y_1) \star (z_0 \star (z_1 \star y_0))$ and $t = ((z_2 \star y_1) \star y_2) \star (z_3 \star y_2)$. Matching the notation of the coming theorem, we use superscripts of u and d (for "up"and "down") to label the distinct occurrences of variables in the cycle, as shown in Table 1.

index	occurrence	term	path	p_i	q_i
0	y_0^u	s	ll	ll	
	y_1^d	t	llr		r
1	y_1^u	s	lr	lr	
	y_2^d	t	lr		Λ
2	y_2^u	t	rr	rr	
	y_0^d	s	rrr		r

Table 1. Occurrences in a cycle.

In the cycle, y_0 is strictly above y_1, since the occurrence y_0^u in s has path ll, which is an initial substring of llr, the path in t of the occurrence y_1^d. And y_1 is above (but not strictly above) y_2, since the occurrence y_1^u in s has path lr, which is a (non-proper) initial substring of lr, the path in t of the occurrence y_2^d. Finally, y_2 is strictly above y_0, since the occurrence y_2^u in t has path rr, which is an initial substring of rrr, the path in s of the occurrence y_0^d.

The theorem also defines relations $\sim$ and $\approx$ on the index set, which is $I = \{0, 1, 2\}$ in our example. We have $1 \sim 2$, since y_1^u is not strictly above y_2^d. The relation $\approx$ is the equivalence relation generated by $\sim$, so its classes are $\{0\}$ and $\{1, 2\}$. The function f that takes each $i \in I$ to the least element in its $\approx$ class has $f(0) = 0$ and $f(1) = f(2) = 1$. Finally, the operation $\star'$ is $\|3, ll, 0\| + \|4, lr, 1\| + \|4, rr, 2\| + \|4, r, 3\|' + \|3, r, 4\|$. The reader can verify that this operation makes $s[0]+s[1]+s[2] = y_0[3]+y_1[4]+(z_1 \star' y_0)[4] = y_0[3]+y_1[4]+y_0[3] = y_1[4]$, where the last step follows since we are adding values modulo 2. Similarly, $t[0]+t[1]+t[2] = (z_2 \star' y_1)[3]+y_2[4]+y_2[4] = y_1[4] + 1 + y_2[4] + y_2[4] = y_1[4] + 1$, which always has a different value.

Theorem 4.2. *Let s and t be groupoid terms which have a cycle. Then s and t are separated in a finite groupoid.*

P r o o f. Let s and t be terms with a cycle as above. So we have a sequence of variables $y_0, y_1, \ldots, y_{m-1}$ where y_0 occurs above y_1, y_1 occurs above y_2, and so on, ending with y_{m-1} occurring above y_0. We may assume that this cycle has minimal length k for all cycles of s and t, and that $k \geqslant 2$ since cycles of length 1 are covered by Theorem 3.1. This implies that all of the variables y_i are distinct. We also adopt the convention that our subscripts are calculated modulo k, so that y_k is the same as y_0.

Each of the y_i has two occurrences in the cycle. For each i, let y_i^u be the occurrence of y_i that is above an occurrence of y_{i+1}, and let y_i^d be the occurrence of y_i that is below an occurrence of y_{i-1}. A given occurrence y' of a variable may be either in the term s or in the term t.

We denote whichever of s and t an occurrence y' is in by $\mathrm{term}(y')$. We will then write $\mathrm{path}(y')$ to denote the path of y' in $\mathrm{term}(y')$. Note that $\mathrm{term}(y_i^u) \neq \mathrm{term}(y_{i+1}^d)$ for all i, since $\mathrm{path}(y_i^u)$ is an initial substring of $\mathrm{path}(y_{i+1}^d)$ and $y_i \neq y_{i+1}$.

We will denote $\mathrm{path}(y_i^u)$ by p_i. And since $\mathrm{path}(y_{i+1}^d)$ has p_i as an initial substring, we will write it as the concatenation $p_i q_i$, where q_i is possibly Λ.

We claim that none of the p_i is an inital substring of any of the others. For contradiction, suppose $i \neq j$ and p_i is an initial substring of p_j. Since y_j^u corresponds to a leaf of $T(\mathrm{term}(y_j^u))$, we must have $\mathrm{term}(y_i^u) \neq \mathrm{term}(y_j^u)$. Now consider y_{j+1}^d. We have that $\mathrm{term}(y_{j+1}^d) \neq \mathrm{term}(y_j^u)$, so $\mathrm{term}(y_{j+1}^d) = \mathrm{term}(y_i^u)$. We also have that p_i

is an initial substring of p_j, which is an initial substring of $p_j q_j = \text{path}(y^d_{j+1})$. In $T(\text{term}(y^u_i))$, this would place the leaf corresponding to the occurrence y^d_{j+1} below the leaf corresponding to y^u_i. The only way this could happen is if $y^u_i = y^d_{j+1}$. So $i = j+1$, and $y^u_i = y^d_i$. But then $\text{term}(y^u_{i-1}) \neq \text{term}(y^d_i) = \text{term}(y^u_i) \neq \text{term}(y^d_{i+1})$, so y^u_{i-1} and y^d_{i+1} are occurrences in the same term. Now $\text{path}(y^u_{i-1})$ is an initial substring of $\text{path}(y^d_i) = \text{path}(y^u_i)$, which is an initial substring of $\text{path}(y^d_{i+1})$, implying that both $\text{path}(y^u_{i-1})$ and $\text{path}(y^d_{i+1})$ label the same leaf of the tree they are in. So $i-1 = i+1 = j$, and our cycle consists of just y_i and y_j, with $y^u_i = y^d_i$ and $y^u_j = y^d_j$. This is a contradiction, since at least one variable occurrence in a cycle must be strictly above the next occurrence. The claim is established.

Without loss of generality, assume that the occurrence y^u_0 is strictly above y^d_1, so $\text{path}(y^d_1)$ is $p_0 q_0$ where $q_0 \neq \Lambda$. Let $I = \{0, 1, 2, \ldots, k-1\}$ be our set of indices for the y_i, and let N be $\{i \in I\colon\ q_i \neq \Lambda\}$. So $0 \in N$.

Define the relation $\sim$ on I by $i \sim j$ if and only if $j = (i+1) \bmod k$ and $q_i = \Lambda$, and let $\approx$ be the equivalence relation generated by $\sim$. Intuitively, the classes of $\approx$ are runs of consecutive indices, with each class ending at an element of N.

Finally, define $f\colon\ I \to I$ by letting $f(i)$ be the least element of the $\approx$ equivalence class of i. This gives us that $f(i) = f(i+1)$ when $q_i = \Lambda$. (We usually have $f(i) \neq f(i+1)$ when $q_i \neq \Lambda$. The one exception is when only one of the q_j is not Λ, so i and $i+1$ are related by $\approx$ the long way around the cycle.)

Now we define the groupoid operation $\star$ to be the sum $[\|k+f(0), p_0, 0\| + \|k+f(1), p_1, 1\| + \ldots + \|k+f(k-1), p_{k-1}, k-1\|] + \sum_{i \in N} \|k+f(i+1), q_i, k+f(i)\|$. Then the operation $\star'$ will be $\star + \|k+f(1), q_0, k+f(0)\|' - \|k+f(1), q_0, k+f(0)\|$, a slight variation of $\star$ where the operation $\|k+f(1), q_0, k+f(0)\|$ is replaced with the tweaked operation $\|k+f(1), q_0, k+f(0)\|'$, while all of the other operations remain unchanged.

We will show that in the groupoid with operation $\star$, the sum modulo 2 of $s[0] + s[1] + \ldots + s[k-1]$ will always equal the sum modulo 2 of $t[0]+t[1]+\ldots+t[k-1]$. Then we will confirm that in the groupoid with operation $\star'$, the two corresponding sums of components will differ. This difference will be caused by the tweaked operation $\|k+f(1), q_0, k+f(0)\|'$, which will only produce an effect in the final output in $\text{term}(y^d_1)$, the term where the occurrence y^d_1 lies. For the moment, we will be working with the operation $\star$.

First, we establish that for any i, the value of the i-th component of $\text{term}(y^u_i)$ will be $y_i[k+f(i)]$. Without loss of generality, let $\text{term}(y^u_i)$ be s. The only summand of $\star$ that assigns a value to $s[i]$ is $\|k+f(i), p_i, i\|$, so $s[i]$ will have the value it assigns. We apply Lemma 3.1, and get that $s[i]$ is equal to $r[k+f(i)]$, where r is the subterm of s with path p_i. In this case, $r = y_i$, so $s[i] = r[k+f(i)] = y_i[k+f(i)]$, as desired.

Given any i, we let $j = i + 1 \bmod k$. We now show that for any y_j, the value of the i-th component of $\mathrm{term}(y_j^d)$ is also $y_j[k + f(j)]$. Without loss of generality, let $\mathrm{term}(y_j^d)$ be t. As before, $t[i]$ will be equal to $w[k + f(i)]$, where w is the subterm of t with path p_i. We now have two cases. If $i \notin N$, then $q_i = \Lambda$ and $\mathrm{path}(y_j^d) = p_i$, making $w = y_j$ and $t[i] = y_j[k + f(i)] = y_j[k + f(j)]$ since $i \sim j$. So assume $i \in N$. Then w is a nontrivial subterm of t, where $\mathrm{path}_w(y_j^d)$ is q_i. The only term in $\star$ that assigns a value to $w[k + f(i)]$ is $\|k + f(i + 1), q_i, k + f(i)\|$, so $w[k + f(i)]$ is $y_j[k + f(i + 1)] = y_j[k + f(j)]$, as desired.

For each i, we do not know which of s and t the occurrences y_i^u and y_{i+1}^d are in. This turns out not to be an obstacle, since we do know that y_i^u and y_{i+1}^d occur in different terms. Working modulo 2, we have that $s[0] + s[1] + \ldots + s[k - 1] + t[0] + t[1] + \ldots + t[k - 1] = [y_0[k + f(0)] + y_1[k + f(1)] + \ldots + y_{k-1}[k + f(k - 1)]] + [y_1[k + f(1)] + y_2[k + f(2)] + \ldots + y_k[k + f(k)]]$, where the second group on the right hand side comes from the y_j^d. But the latter expression is equal to $2[y_0[k+f(0)]+\ldots+ y_{k-1}[k + f(k - 1)]] = 0$ modulo 2. Since $s[0] + \ldots + s[k - 1]$ and $t[0] + \ldots + t[k - 1]$ sum to 0, they have the same parity.

Now we turn to the groupoid with operation $\star'$, and consider the effect of the tweaked operation $\|k + f(1), q_0, k + f(0)\|'$. The reader may verify that everything works as before, except in the calculation of the 0-th component of $\mathrm{term}(y_1^d)$. As before, we may assume that s is $\mathrm{term}(y_1^d)$. We then get $s[0] = r[k + f(0)]$, where $r[k + f(0)]$ is found using $\|k + f(1), q_0, k + f(0)\|'$. This makes $r[k] = y_1[k + f(1)] + 1 \bmod 2$, giving $s[0] = y_1[k + f(1)] + 1 \bmod 2$. This in turn changes the parity of $s[0] + s[1] + \ldots + s[k - 1]$ in whichever term we are calling s, as desired.

As in Theorem 3.1, this yields a finite groupoid that separates s and t. □

There are pairs of terms without a cycle which still cannot be unified. For example, let $s = (x \star y) \star (z \star y)$ and let $t = z \star ((y \star y) \star (x \star x))$. Working left to right, we see that x and y occur below z, y occurs below z and x occurs below y. This is consistent with the ordering $x < y < z$. Since there is a consistent ordering of the variables like this, there are no cycles. However, s and t cannot be unified. If we attempt to construct the unification closure $\equiv$ of s and t, we get $z \equiv y \star y$, $z \equiv y \star y$ and $y \equiv x \star x$ by repeated applications of the unification axiom. Then $x \star y \equiv z \equiv y \star y$, and so $x \equiv y$ by the unification axiom. Thus $x \equiv y \equiv x \star x$, and $\equiv$ cannot be acyclic in the sense of Definition 4.1.

Although Theorem 4.2 does not apply to this last example, we had no problem separating the terms using a similar construction. Letting

$$\star' = \|3, l, 0\| + \|3, rl, 1\| + \|4, rr, 2\| + \|4, l, 3\| + \|4, l, 4\|',$$

we calculate

$$s[0] + s[1] + s[2] = (x \star' y)[3] + z[3] + y[4] = x[4] + z[3] + y[4],$$

while

$$t[0] + t[1] + t[2] = z[3] + (y \star' y)[3] + (x \star' x)[4] = z[3] + y[4] + x[4] + 1,$$

which has the opposite parity.

Based on many examples similar to the above, we make the following conjecture.

Conjecture 4.1. Whenever two groupoid terms can be separated in an infinite groupoid, they can also be separated in a finite groupoid.

References

[1] *F. Baader, W. Snyder*: Unification theory. Handbook of Automated Reasoning (A. Robinson et al., eds.). North-Holland/Elsevier, Amsterdam, MIT Press Cambridge, 2001, pp. 445–533. zbl doi

[2] *M. S. Braitt, D. Hobby, D. Silberger*: Completely dissociative groupoids. Math. Bohem. *137* (2012), 79–97. zbl MR

[3] *M. S. Braitt, D. Silberger*: Subassociative groupoids. Quasigroups Relat. Syst. *14* (2006), 11–26. zbl MR

[4] *S. Burris, H. P. Sankappanavar*: A Course in Universal Algebra. Graduate Texts in Mathematics 78, Springer, New York, 1981. zbl MR

[5] *A. Drápal, T. Kepka*: Sets of associative triples. Eur. J. Comb. *6* (1985), 227–231. zbl MR doi

[6] *J. Herbrand*: Recherches sur la théorie de la démonstration. Travaux de la Société des Sciences et des Lettres de Varsovie *33* (1930), 128 pages. (In French.) zbl MR

[7] *G. P. Huet*: Résolution d'équations dans des langages d'ordre $1, 2, \ldots, \omega$. Thèse d'État, Université de Paris VII (1976). (In French.)

[8] *J. Ježek, T. Kepka*: Medial groupoids. Rozpr. Cesk. Akad. Ved, Rada Mat. Prir. Ved *93* (1983), 93 pages. zbl MR

[9] *D. E. Knuth*: The Art of Computer Programming. Vol. 1: Fundamental Algorithms. Addison-Wesley Series in Computer Science and Information Processing, Addison-Wesley, London, 1968. zbl MR

[10] *J. A. Robinson*: A machine-oriented logic based on the resolution principle. J. Assoc. Comput. Mach. *12* (1965), 23–41. zbl MR doi

[11] *R. P. Stanley*: Enumerative Combinatorics. Vol. 2. Cambridge Studies in Advanced Mathematics 62, Cambridge University Press, Cambridge, 1999. zbl MR doi

Authors' addresses: *Milton Braitt*, Universidade Federal de Santa Catarina, R. Eng. Agronômico Andrei Cristian Ferreira, Trindade, Florianópolis, Santa Catarina 88040–900, Brazil, e-mail: `m.braitt@ufsc.br`; *David Hobby*, *Donald Silberger*, State University of New York, 1 Hawk Drive, New Paltz, NY 12561, USA, e-mail: `hobbyd@newpaltz.edu`, `silbergd@newpaltz.edu`.